Recetario de LaTeX

2002-2006 por Aristarco, http://aristarco.com.es

Índice

Introducción

No voy a hacer aquí una discusión tipo LaTeX vs MS Word, Corel WordPerfect o Sun StarOffice. Opino que cada una de las filosofías de trabajo que conllevan estas opciones tiene sus puntos positivos y negativos; sin embargo, es una ventaja conocer ambas, pues ello nos permite seleccionar la más apropiada para cada caso concreto. Dicho de forma breve, los procesadores de texto WYSIWYG (*What You See Is What You Get*) tales como los mencionados más arriba resultan bastante cómodos para escribir documentos con texto y gráficos pero que no sean demasiado largos; la ventaja aquí es que el resultado final se muestra en pantalla en todo momento. Sin embargo, para proyectos grandes y que necesiten de un acabado profesional, LaTex ofrece mayor control, estabilidad y robustez, sin embargo, el usuario trabaja con un documento en el que aparecen comandos LaTeX (de hecho, LaTeX se puede considerar un lenguaje de programación), y que no se parece al aspecto final que tendrá el documento una vez tratado por LaTeX.

El principal problema de todo principiante a LaTeX (yo mismo en estos momentos) es la pendiente inicial de la curva de aprendizaje. Uno se pone delante de un procesador WYSIWYG y puede hacer multitud de cosas casi al momento, con tan sólo probar un poco y buscar entre los menús. Con relativamente poco esfuerzo uno puede llegar a tener un dominio del entorno suficiente como para afrontar la mayoría de situaciones, o incluso todas ellas, dependiendo de nuestras necesidades como usuarios. Para usos avanzados hay que trastear un poco más, buscar en la ayuda en línea o consultar un manual. Sin embargo, en LaTex, uno no puede sentarse y juguetear a ver qué sale. Hay que estudiar primero y probar después. Uno no puede crear un documento estúpidamente sencillo que contenga el típico "hola, mundo" sin tener que estudiar primero una serie de comandos LaTeX. Al principio se tiene la sensación de no poder controlar casi nada. El aspecto de los primeros documentos es estático y aburrido, y uno empieza a ver que para hacer cosas más vistosas tendrá que dedicar mucho, mucho tiempo. Además, LaTeX no dispone de un entorno de usuario, simplemente se usa un editor de textos básico y luego el fichero resultante se hace pasar por el programa LaTeX, que se ejecuta desde la línea de comandos (existen algunos editores especiales para LaTeX que facilitan un poco el trabajo, ver más adelante). Sin embargo, después de paciencia y práctica, empiezan a verse claras las ventajas. Es muy difícil que LaTeX falle o se quede colgado (todavía no he visto nada de esto después de todas las veces que lo he usado). Pero además, si lo hiciese, nuestro fichero fuente ya estaba salvado previamente, no perderemos el trabajo. Se trabaja cómodamente con múltiples ficheros, y el autonumerado de capítulos, secciones, índices, tablas, figuras, ecuaciones, etc. funciona perfectamente, ahorrándonos muchos quebraderos de cabeza, especialmente cuando extraemos capítulos o secciones de un documento y lo insertamos en otro distinto (es copiar y pegar, literalmente). Y si uno trabaja con frecuencia con ecuaciones, es casi obligado emplear LaTeX.

Este documento, dentro de mis posibilidades como principiante en la materia, trata de ayudar a todo aquél que desee iniciarse en LaTeX. Para ello, se proporcionan aquí una serie de recetas básicas para empezar a obtener buenos resultados con LaTeX desde el primer encuentro. Por desgracia, para iniciarse hace falta leer un poco antes de referirse a estas recetillas. Podría contar todo lo necesario aquí, de forma resumida, pero por el momento no dispongo del tiempo suficiente... Quizá algún día.

Primeros pasos en LaTeX

Como decía, antes de empezar con las recetas hay que leer un poco. Resulta que hay por ahí gente competente que hace bien las cosas, y además, las hace gratis... Para disipar posibles dobles interpretaciones, añadiré que me refiero en este caso a "**The Not So Short Introduction to LaTeX**" (en formato PDF, usar Adobe Acrobat Reader o Adobe Acrobat). Se trata de un libro electrónico de libre distribución, fácil de leer y no muy gordo, que constituye el material básico para empezar. Mi recomendación es descargarlo, imprimirlo, y a medida que uno lee, ir estudiando la plantilla de documento LaTeX que se incluye en el apartado 4. La plantilla se puede copiar y pegar, y se pueden hacer pruebas fácilmente con ella. Hay una traducción al español de ese texto, aunque quizá de una versión anterior.

Hay que decir, como culturilla general "latexera", que los libros definitivos sobre LaTeX son "**LaTeX: A Document Preparation System**" de Leslie Lamport, el padre de LaTeX, y "**The LaTeX Companion**", Goosens, M. et al. Es frecuente ver referencias a ambos en cualquier documentación que consultemos acerca de LaTeX.

Apúntate por ahí que el sitio web definitivo sobre LaTeX es CTAN (Compehensive TeX Archive Network). Cuando te haga falta profundizar en algo que no encuentres en el libro introductorio o en estas recetas, puedes echar un vistazo ahí. Las distribuciones de LaTeX suelen instalar en el disco

duro muchos archivos de ayuda, pero la forma en que están organizados no suele facilitar mucho las cosas; uno tiene que saber de antemano el nombre del paquete que necesita para no perderse. Si sigues sin encontrar lo que andas buscando, siempre puedes probar con Google, aunque a veces no sabe uno muy bien qué términos de búsqueda poner para encontrar lo que necesita. Como no podía ser menos, LaTeX tiene presencia en los grupos de noticias de Usenet **comp.text.tex** (general, en inglés) y **es.comp.lenguages.tex** (español) así que frecuentemente uno obtiene respuestas a sus dudas simplemente buscando entre los mensajes allí presentes; pero si esto no es fructífero, siempre se puede recurrir al viejo método de preguntar. Si uno es educado, y no pregunta tonterías, le responden en el día. Google proporciona acceso web a estos grupos si no tienes acceso a un servidor de news, Google Groups comp.text.tex y Google Groups es.comp.lenguajes.tex; de todos modos, es recomendable usar Google Groups ya que, a diferencia de los clientes de noticias, éste busca también en el cuerpo de los mensajes, no sólo en el campo de asunto. Existen asociaciones de usuarios de LaTeX, en España funciona CervanTeX, no dejes de echar un vistazo a su página web y en particular, a su sección de preguntas frecuentes.

Ahora nos hace falta decidir qué entorno vamos a usar para escribir nuestros documentos LaTeX.

Entornos de desarrollo en LaTeX

Desde el punto de vista del usuario, LaTeX se presenta como una programa de línea de comandos que toma como parámetro principal el fichero fuente que contiene la descripción (texto y comandos LaTeX) del documento a generar. Existen dos comandos para ejecutar LaTeX:

- **`latex`** genera el documento final en formato **DVI** (*DeVice Independent*), a partir del cual puede obtenerse, mediante la aplicación `dvips`, el documento en formato PS (*PostScript*):

  `latex fichero.tex` (genera fichero.dvi, necesita un visualizador específico, pero por la naturaleza del formato, la lectura resulta lenta)
  `dvips fichero.dvi -o fichero.ps` (genera fichero.ps, más manejable que DVI y directamente entendible por muchas impresoras láser, se puede ver con GhostScript/GhostView)

- **`pdflatex`** genera el documento directamente en formato **PDF** (*Portable Document File*, de uso muy extendido en Internet):

  `pdflatex fichero.tex` (genera fichero.pdf, visualizable con Adobe Acrobat Reader)

A la hora de incluir gráficos o imágenes en los documentos, hay que tener en cuenta que cada una de estas aplicaciones es capaz de comprender sólo unos ciertos formatos gráficos; si necesitásemos incrustar figuras almacenadas en otros formatos no directamente soportados, tendremos que recurrir a un programa que haga la conversión.

- `latex` trabaja cómodamente tan sólo con EPS (Encapsulated PostScript, una variante especial de PS).
- `pdflatex` espera que las figuras estén en PDF (preferible para los gráficos vectoriales), PNG (adecuado para las capturas de pantalla o cualquier imagen raster generada por computador) o JPG (adecuada para fotografías).

"Vale, pero, ¿cómo instalo esos comandos en mi sistema operativo?" LaTeX puede usarse en Linux (y otros sistemas tipo UNIX) y en MS Windows (aunque parezca sorprendente). Dependiendo del sistema operativo, la distribución y el método de instalación varía.

LaTeX en Linux

Hay que instalar el paquete `tetex`. Éste incluye todo lo necesario, excepto el editor para poder escribir los documentos LaTeX.
Como editor puede usarse:

- Emacs (paquete `emacs`), que dispone de un modo de edición especial para LaTeX, realzando los comandos. Puede ser conveniente evaluar una extensión para emacs denominada AUCTeX, que indenta automáticamente, entre otras cosas.
- LyX (paquete `lyx`), entorno integrado que nos permite editar gráficamente el documento al proporcionar una visualización esquemática que ayuda a hacerse una idea del aspecto final.
- Ktexmaker2, para KDE, que es un entorno integrado con aspecto parecido al de las herramientas de programación visual, permitiendo un acceso cómodo a los distintos ficheros de un proyecto LaTeX, así como a cada uno de sus elementos (secciones, figuras, tablas, etc). El editor dispone de realce de comandos. Integra el acceso al previsualizador DVI y a Gnuplot (que genera gráficas 2D y 3D).

Los gráficos e imágenes, en Linux, normalmente se imprimen como PostScript. Derivando la salida de impresión a fichero, disponemos de la figura en formato PS. Para generar el EPS/PDF correspondiente puede usarse GhostScript/GhostView.

LaTeX en MS Windows

Debe instalarse la distribución MiKTeX. Realmente se instala una versión reducida (guiada por asistente al típico estilo Windows), pero luego se pueden descargar aquellos módulos LaTeX (denominados **paquetes**) que se vayan necesitando mediante *MiKTeX Options*, que viene incluido en la distribución. La distribución de MiKTeX se puede ir actualizando mediante *MiKTeX Update Wizard*.

Como editor puede usarse TeXnicCenter, que es muy similar a Ktexmaker2 de Linux, y también de libre distribución (aunque todavía en desarrollo, es bastante funcional). Una alternativa sería WinEDT, aunque yo no la recomendaría; es shareware y además da algunos problemas. Una vez instalado TeXnicCenter, hay que realizar unos sencillos pasos de configuración; básicamente confirmar la ruta de acceso a MiKTeX e indicar que se va a usar PDF (preferible, aunque también puede ser DVI o PS) como formato de salida, esto hará que TeXnicCenter ejecute automáticamente el comando LaTeX apropiado. De este modo, una vez escrito el documento LaTeX, para generar la salida tan sólo tendremos que pulsar el icono "Build" y observar la ventana de mensajes de error. TeXnicCenter proporciona botones para navegar por los errores, avisos generales (*warnings*) y "*bad boxes*" (avisos `overfull \hbox` y `underfull \hbox`). Si hubo éxito, se puede visualizar el fichero generado pulsando el icono de previsualización. En definitiva, a mí me ha ido bien con él.

Los gráficos vectoriales, en Windows, suelen presentarse en formato WMF o EMF (*Windows MetaFile*/*Enhanced MetaFile*). Es necesario convertirlos a EPS ó PDF dependiendo de si queremos que el documento de salida esté en PS ó PDF, respectivamente.

- La conversión de un gráfico WMF/EMF a EPS puede hacerse con WMF2EPS, que necesita para funcionar el driver de impresora PS de Adobe (la ayuda de WMF2EPS explica con todo detalle los pasos para su instalación). El gráfico original puede copiarse desde la aplicación con la que se ha construido, y pegarse en WMF2EPS, el cual generará la versión EPS del mismo.
- La conversión de un gráfico WMF/EMF a PDF también requiere de WMF2EPS, pero además hay que usar un filtro EPS a PDF:
  - GhostScript/GhostView, que toma el fichero EPS y lo convierte a PDF;
  - Adobe PDFWriter, que se integra con WMF2EPS, permitiendo obtener la salida directamente en PDF sin tener que generar el EPS intermedio;
  - Adobe Distiller, que toma el fichero EPS y lo convierte a PDF.

La inclusión de gráficos vectoriales, por lo anterior, resulta un tanto incómoda, al requerir la conversión de todos y cada uno de los gráficos, mediante un proceso que requiere varios pasos. Esto resulta algo tedioso y lento. Por el momento, no conozco solución mejor que ésta.

Las imágenes raster o fotográficas, independientemente del formato utilizado, son fácilmente convertibles a PNG o JPG (mediante XnView o IrfanView, por ejemplo) respectivamente, y con estos formatos podemos directamente emplear `pdflatex`.

Algunos trucos para el uso de MiKTeX y TeXnicCenter

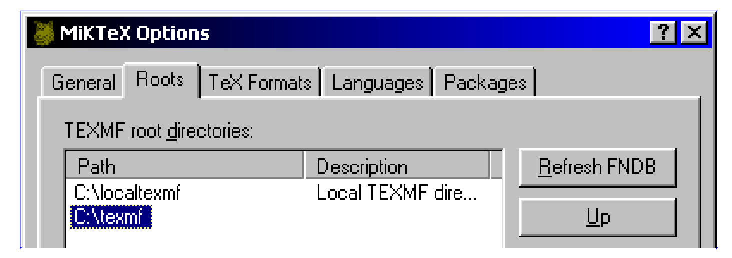
Instalación automatizada de nuevos paquetes en MiKTeX:

MiKTeX dispone de una aplicación de mantenimiento de paquetes que se instala con la distribución. Se accede mediante la carpeta "MiKTeX" del botón de Inicio de Windows y se denomina "*MiKTeX Options*". Para instalar un paquete, se ejecuta esta aplicación, se selecciona la pestaña *Packages*, y pulsando el botón derecho sobre *MiKTeX Packages* aparece la opción *Search*. Al introducir el nombre del paquete a instalar, la lista de paquetes se despliega mostrando el paquete solicitado, que llevará asociada una casilla de verificación que debemos seleccionar para instalar. A continuación, se pulsa en "Aceptar" o "Aplicar" y el paquete se descargará e instalará automáticamente en la distribución, listo para su uso.

Instalación manual de nuevos paquetes en MiKTeX:

Cuando algún paquete no está disponible en la distribución de MiKTeX o falla la instalación automática (lo cual no es nada frecuente), podemos recurrir a descargar el paquete desde CTAN y luego instalarlo a mano. Lo primero será descomprimirlo, lo cual nos proporcionará, al menos, un archivo con extensión DTX. Este archivo contiene tanto el código fuente del paquete como la documentación asociada. Para obtener ambas cosas hay que hacer lo siguiente:

1. ejecutar `latex` *paquete*`.dtx` desde la consola de MS-DOS (MiKTeK, al instalarse, debió establecer el path de MS-DOS apropiadamente). Puede hacer falta ejectuar este comando hasta tres veces para resolver todas las referencias cruzadas. Este paso genera, entre otros, los archivos *paquete*`.ins` y *paquete*`.dvi`; (*)
2. ejecutar `latex` *paquete*`.ins` para obtener *paquete*`.sty`, que es el fichero que hay que copiar en `c:\texmf\tex\latex\`*paquete*`\`. Ahora hay que ejecutar *MiKTeX Options*, seleccionar la pestaña *Roots*, directorio `c:\texmf`, y pulsar *Refresh FNDB*; con esto el paquete ya puede usarse desde cualquier documento latex



3. ejecutar `dvipdfm` *paquete*`.dvi` para obtener *paquete*`.pdf`, que contiene la documentación asociada al paquete; debe copiarse al directorio `c:\texmf\doc\latex\`*paquete*`\`
4. Borrar el resto de ficheros, ya no son necesarios (incluídos los originales)

`(*)` A veces, al obtener los fuentes del paquete ya se dispone del fichero *paquete*`.ins`; si no se necesita la documentación, la instalación del paquete se reduce al paso 2.

Adaptación de LaTeX al español

Una de las cosas que el usuario de habla hispana se encuentra con LaTeX es que genera algunas cosas en inglés, tales como "table of contents" o "chapter". En "The Not So Short Introduction to LaTeX" no se dice nada de cómo cambiar esos títulos generados automáticamente. ¿Y qué pasa con las **tildes** o las e**ñ**es?

No hay problema. Uno puede escribir el documento LaTeX usando esos caracteres, aunque hay que incluir lo siguiente tras `\documentclass`:

```
\usepackage[latin1]{inputenc}
```

Para cambiar los títulos de capítulos, secciones, índices, etc. se usan los siguientes comandos (puede que no hagan falta todos), que deben colocarse tras `\begin{document}`:

```
\newcommand\contentsname{Índice general}
\newcommand\listfigurename{Lista de Figuras}
\newcommand\listtablename{Lista de Tablas}
\newcommand\bibname{Bibliografía}
\newcommand\indexname{Indice alfabético}
\newcommand\figurename{Figura}
\newcommand\tablename{Tabla}
\newcommand\partname{Parte}
\newcommand\chaptername{Capítulo}
\newcommand\appendixname{Apéndice}
\newcommand\abstractname{Resumen}
```

Todos estos comandos se han incluido en la plantilla aquí presentada.

Otra importante adaptación del entorno LaTeX al castellano consiste en configurar éste para que aplique los patrones de guionado (*hyphenation*) propios del español a la hora de partir palabras en distintas líneas. En MiKTeX esto se consigue fácilmente gracias a la herramienta *MiKTeX Options*:

1. Seleccionar la pestaña *Languages*
2. Seleccionar el idioma "spanish" para documentos en español, "english" para documentos en inglés, etc., **deseleccionando todas las demás opciones**. Sólo debe quedar una activa. Si lo que se desea es evitar en lo posible la ruptura de palabras, seleccionar "nohyphenation"

en la lista de idiomas.
3. Seleccionar la pestaña *General*
4. Pulsar botón *Update Now* en la sección *Format files*. A consecuencia de esto, MiKTeX puede indicar que no puede crear algunos ficheros de formato, esto es normal,
ya que la instalación estándar de MiKTeX no es la completa.
5. Por último, pulsar *Aceptar* y la ventana de MiKTeX se cierra. La próxima vez que se ejecute LaTeX se usará el patrón de silabeo seleccionado, sin necesidad de añadir nada al código LaTeX.

Plantilla de documento LaTeX comentada

El fichero de plantilla es el que se muestra a continuación, comentado. Conviene ir revisándolo a medida que avanzamos con "The Not So Short Introduction to LaTeX". Esta plantilla tiene la particularidad de que corresponde al estilo libro (*book*), que es el más completo de los estilos que se comentan ahí. Los demás son una versión reducida de éste.

Inicio del documento:

- Definición del tipo de documento (aquí, "book")
- Inclusión de paquetes
- Definición de título, autores, fecha; estos campos se usan para generar una página de portada con el comando \maketitle; se puede cambiar para incluir una portada gráfica más elaborada
- Otros comandos de configuración
- Estilo de encabezados y pies de página (ver técnicas más avanzadas)

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Nombre del trabajo,
% título,
% autores
% fechas,
% comentarios, etc.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

\documentclass[a4paper,twoside,11pt]{book}

% Incluir los paquetes necesarios
\usepackage[latin1]{inputenc} % Caracteres con acentos.
\usepackage{latexsym} % Símbolos
\usepackage{graphicx} % Inclusión de gráficos. Soporte para \figura (más
abajo)
\usepackage[pdftex=true,colorlinks=true,plainpages=false]{hyperref} %
Soporte hipertexto

% Título, autor(es), fecha.
\title{\Huge \textbf{Título}}
\author{\huge \textit{Autor 1} \and \huge \textit{Autor 2}}
\date{\Large Mes, Año}

\sloppy % suaviza las reglas de ruptura de líneas de LaTeX
\frenchspacing % usar espaciado normal después de '.'
\pagestyle{headings} % páginas con encabezado y pie básico
```

Definición de macros, aquí concretamente se incluyen comandos para la insertar cómodamente figuras y tablas. Sin embargo, el uso de estos comandos hace que Ktexmaker2 y TeXnicCenter no sean capaces de identificar la posición de figuras y tablas en el documento. La alternativa sería, por tanto, usar esos mismos fragmentos de código LaTeX y copiarlo y pegarlo cada vez que haya que incrustar una figura o tabla.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Comando:
% \figura{nombre-fichero}{argumentos}{título}{etiqueta}
% Resultado:
% Inserta una figura. "La figura~\ref{etiqueta} muestra..." permite
% referenciar la figura desde el texto.
% argumentos: width=Xcm,height=Ycm,angle=Z
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
\newcommand{\figura}[4]{

     \begin{figure}
     \begin{center}
     \includegraphics[#2]{#1}
     \caption{#3}
     \label{#4}
     \end{center}
     \end{figure}

}

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Entorno:
% \begin{tabla}{título}{etiqueta}
% ... (contenido tabla)
% \end{tabla}
% Resultado:
% Inserta una tabla.
% Contenido tabla se define mediante un entorno 'tabular'.
```

```
% "Tabla~\ref{etiqueta}" permite referenciar la tabla.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
\newenvironment{tabla}[2]{

     \begin{table}
     \begin{center}
     \caption{#1}
     \label{#2}

} {

     \end{center}
     \end{table}

}
```

Comienzo de la sección del documento

```
\begin{document} % Inicio del documento
```

Renombrado de secciones para que muestren el título correcto en español (asunto discutido en el apartado sobre adaptación al español)

```
\renewcommand{\contentsname}{Indice}
\renewcommand{\partname}{Parte}
\renewcommand{\chaptername}{Capítulo}
\renewcommand{\appendixname}{Apéndice}
\renewcommand{\bibname}{Bibliografía}
\renewcommand{\figurename}{Figura}
\renewcommand{\listfigurename}{Indice de figuras}
\renewcommand{\tablename}{Tabla}
\renewcommand{\listtablename}{Indice de tablas}
```

Materias iniciales del libro, incluyendo:

- Página de título o portada
- Prólogo
- Tabla de contenidos
- Índice de figuras
- Índice de tablas

```
\frontmatter % Incluye: título, prólogo, índices

\maketitle % Título

\chapter{Prólogo} % Prólogo
% ... (contenido del prólogo)

% inserta todos los epígrafes hasta el nivel \paragraph en la tabla de
contenidos
\setcounter{tocdepth}{4}
% numera todos los epígrafes hasta nivel \subparagraph en el cuerpo del
documento
\setcounter{secnumdepth}{5}

\tableofcontents % Tabla de contenido
\newpage
\listoffigures % Índice de figuras
\newpage
\listoftables % Índice de tablas
\newpage
```

Materias principales del libro, que incluye los capítulos y apéndices. Puede subdividirse en "partes" mediante el comando `\part`, donde cada parte contendrá uno o más capítulos.

Un capítulo puede escribirse en un fichero distinto, digamos "fichero.tex", el cual se incluiría desde aquí mediante el comando `\include{fichero}`

```
\mainmatter % Capítulos

\chapter{Capítulo A}
\label{ch:capituloA}
% ... (contenido del capítulo A)

\chapter{Capítulo B}
\label{ch:capituloB}
% ... (contenido del capítulo B)

\appendix % Apéndices
\chapter{Apéndice X}
\label{ch:apendiceX}
% ... (contenido del apéndice X)

\chapter{Apéndice Y}
\label{ch:apendiceY}
% ... (contenido del apéndice Y)
```

Materias finales del libro, en este caso la bibliografía. Aquí se ha optado por poner la bibliografía en un

```
\backmatter
% Bibliografía
\include{bibliografia}
```

fichero aparte denominado "bibliografia.tex"	
Marca el final del documento	`\end{document} % Fin del documento`

El fichero " bibliografia.tex " tendría el siguiente aspecto:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Bibliografía.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

\begin{thebibliography}{99}

\bibitem{etiqueta-referencia1} Autor: \textit{Título}. Editorial, año.
\bibitem{etiqueta-referencia2} Autor: \textit{Título}. Editorial, año.
% ...
```

En cuanto a la bibliografía, hay que decir que es preferible el uso de BiBTeX en lugar del entorno `thebibliography`. Pronto se incluirán indicaciones para el uso de este potente paquete en este recetario.

Errores y avisos frecuentes de LaTeX

Siendo todavía un novato en LaTeX, no es un tema que yo conozca ni siquiera mínimamente. Lo que sí puedo decir es que la criatura ha salido un poco quejica, pero en el fondo es por nuestro bien. Básicamente (y muy, pero que muy) distinguimos estos tipos de mensajes:

- **Errores**. Implican que la salida no será generada. Normalmente se deben a un olvido de algún '}' o ']' o confusión entre ellos. En estos casos, LaTeX normalmente acierta al indicar la causa del error, pero no suele dar muchas pistas acerca de en qué línea está el problema. En otras ocasiones, el error es más extraño, ahí me pierdo.
- **Avisos**. Son mensajes que indican la existencia de algún problema no fatal, de modo que LaTeX puede seguir analizando el documento y generando la salida (aunque probablemente ésta no sea satisfactoria). Básicamente:
  - **Undefined references** (referencias no definidas):
    `LaTeX Warning: Reference `tab:tmn-std-informacion' on page 234 undefined on input line 4873.`
    Son frecuentes y no necesariamente dañinos. En la primera pasada, LaTeX normalmente no es capaz de resolver todas las referencias, por lo que es necesario dar una segunda pasada (es decir, volver a ejecutar LaTeX sobre el documento). A veces es necesaria una tercera pasada. Cuando es necesario dar una nueva pasada, LaTeX lo dice explícitamente:
    `LaTeX Warning: There were undefined references.`
    `LaTeX Warning: Label(s) may have changed. Rerun to get cross-references right.`
    Sin embargo, otras veces se produce por haber introducido erróneamente la etiqueta en cuestión. En estos casos, después de la segunda/tercera pasada, LaTeX sigue informando del error, así que debemos subsanarlo.

  - **Overfull \hbox**: se produce cuando LaTeX no sabe cómo dividir una palabra que cae al final de una línea, y que, por tanto, invade el margen derecho. Normalmente el resultado no es aceptable; la solución pasa por buscar la palabra en cuestión e indicar cómo hay que romperla. Hay casos más peliagudos, como el ejemplo que se muestra a continuación, en el que el problema se produce en el encabezado. En este caso, romper la palabra no sirve de mucho; habría que plantearse si modificar el formato de el encabezado para permitir dos líneas, si eliminar la palabra "Capítulo", si obligar a LaTeX a usar minúsculas en el encabezado, o por supuesto, cambiar el nombre del capítulo a algo más corto. Nótese que el aviso indica cúanto espacio ha invadido la línea en el margen derecho (`xx.xxxxxpt too wide`); se indica en pt, es decir, en unidades de 1/72 pulgadas. A modo de referencia, cada 10pt son aproximadamente 3,5mm, por lo que el aviso del ejemplo nos indica una invasión del margen derecho de unos 14mm.

    ```
    Overfull \hbox (39.74638pt too wide) has occurred while \output is active
    []/pplr7t@10.95pt/228 /pplro7t@10.95pt/CAP[]ITULO 5. ITU-T TELECOMMUNICATIONS
    MANAGEMENT NETWORK
    []
    ```

    228CAPÍTULO 5. ITU-T TELECOMMUNICATIONS MANAGEMENT NETWORK

    para establecer comunicación con él. En definitiva, se trata de un servicio que mapea nombres en direcciones. Si un sistema tiene esta información almacenada localmente, el servicio de directorio es implícito; la mayor flex-

  - **Underfull \hbox**: se produce cuando, en el proceso de justificación, la línea queda con demasiados espacios en blanco entre palabras. Normalmente no son dañinos, pues el resultado no es demasiado desagradable a la vista. La única solución sería escribir las cosas de otra manera. La gravedad de la situación viene dada por el parámetro "*badness*"; sin embago, incluso para valores altos puede verse que el resultado es aceptable, por ejemplo:

    ```
    Underfull \hbox (badness 10000) in paragraph at lines 1398--1399
    /pplr7t@10.95pt/ti-va-mente. Para definir un tipo sin[]onimo al tipo en-tero:
    []
    ```

4.3.2.1.2 Tipos básicos

Los tipos básicos son aquellos que presentan una estructura atómica, sin componentes internos. Los tipos básicos ASN.1 son los siguientes:

- BOOLEAN. Se trata de un tipo de datos cuyos valores sólo pueden ser TRUE (verdadero) o FALSE (falso). Para definir un tipo sinónimo al tipo booleano: <nombreTipo> ::= BOOLEAN.
- INTEGER. El tipo numérico que incluye a todos los valores enteros posibles, positivos, negativos y cero, sin límite de tamaño[12]. Se han definido los valores especiales PLUS-INFINITY y MINUS-INFINITY para representar $+\infty$ y $-\infty$, respectivamente. Para definir un tipo sinónimo al tipo entero: <nombreTipo> ::= INTEGER.
- REAL. Es el tipo de los números expresados en coma flotante, y por

Cambiar el tipo de letra de un documento

La técnica más sencilla consiste sencillamente en incluir el paquete `font`, del siguiente modo:

`\usepackage{font}`

donde `font = {`**`bookman`**` | `**`newcent`**` | `**`palatino`**` | `**`times`**`}`. Si no se incluye el comando, el tipo es "computer roman".

Para que lo anterior funcione, es necesario instalar en la distribución el paquete **`psnfss`** si no estaba ya incluido.

Encerrar una letra en un círculo

El comando **`\textcircled{`*`x`*`}`** genera el carácter *x* encerrado en un círculo.

Definición de puntos de ruptura en las palabras

Muy importante cuando se generan mensajes "*Overfull \hbox*" al procesar el fichero LaTeX, normalmente debidos a que éste no sabe como romper palabras al cambiar de línea: usar **`\-`**, por ejemplo: `super`**`\-`**`califragilístico`**`\-`**`espialidoso`, que permitiría las siguientes rupturas:

- supercalifragilísticoespialidoso (sin ruptura)
- super-
califragilísticoespialidoso
- supercalifragilístico-
espialidoso

Si la palabra aparece frecuentemente, es preferible usar el comando **`\hyphenation{...}`** en el preámbulo (es decir, antes de `\begin{document}`) para definir el patrón de ruptura una única vez en todo el documento:

```
\hyphenation{super-califragilistico-espialidoso}
```

Para palabras acentuadas es preferible el uso de **`\-`**.

NOTA: las rupturas propuestas en estos ejemplos son poco flexibles y no se corresponden con las reglas del español, pero con ello se pretende hacer notar que se tiene total libertad para determinar los puntos de ruptura.

Si LaTeX está rompiendo mal palabras comunes en el idioma, conviene cerciorarse de que el entorno ha sido configurado convenientemente con los patrones de guionado del idioma en el que se está escribiendo el documento.

Insertar el nombre de una sección en el texto

Dentro del paquete `hyperref` hay un estilo llamado `nameref`, que permite lo siguiente:

```
\section{Nombre de la sección}
\label{sec:nombre}
...
\nameref{sec:nombre} % esto se sustituye por el texto "Nombre de la sección" (con hyperref, además es
hipertexto)
```

Incluir notas (texto, raya vertical o icono) al margen

Se trata de usar el comando `\marginpar[`*texto que va al margen izq*`]{`*texto que va al margen derecho*`}`. En estilos a dos caras, usando `\marginpar{`*texto de la nota al margen*`}` automáticamente se elige el margen más externo (el más ancho, por defecto). El comando tiene algunos problemas, para solucionarlos, se puede definir un nuevo comando:

`\newcommand{\marginalnote}[1]{\mbox{}\marginpar{\raggedright\hspace{0pt}#1}}`

y se emplea del siguiente modo: **`\marginalnote{`***texto al margen***`}`**

Algunos usos especiales:

Raya vertical al lado de un párrafo

```
...
\usepackage{calc}
...
\newcommand{\marginalnote}[1]{\mbox{}\marginpar{\raggedright\hspace{0pt}#1}}

% \rayamargen[color]{ancho}{alto}
% color (opcional): black (valor por defecto), red, blue, ... o cualquier otro definido
% ancho: debe especificarse como longitud (es decir, hay que indicar las unidades)
% alto: altura en número de líneas de texto a ocupar
\newcommand{\rayamargen}[3][black]{%
  \marginalnote{%
    \textcolor{#1}{%
      \rule[\baselineskip * (1-#3)]{#2}{\baselineskip * (#3-1) + 1.5ex}%
    }
  }
}
...
\begin{document}
...
\rayamargen{2mm}{3}La tercera y última parte, \nameref{sec:gestion-telecontrol}, representa
la parte más importante del presente trabajo. El capítulo...
...
\end{document}
```

basada en documentos normalizados acerca de la gestión de redes de telecomunicaciones.

La tercera y última parte, Uso de arquitecturas de gestión de comunicaciones en el telecontrol de redes eléctricas, representa la parte más importante del presente trabajo. El capítulo ?? estudia con detalle la situación actual del telecontrol de las redes eléctricas y así determinar el grado de idoneidad que tendría la adopción de arquitecturas de gestión de redes

Notas sobre un texto subrayado

```
...
\usepackage{calc}
...
\usepackage{color,soul}
...
\newcommand{\marginalnote}[1]{\mbox{}\marginpar{\raggedright\hspace{0pt}#1}}

% \notamargen[color]{texto_subrayado}{texto_al_margen}
% color (opcional): blue (valor por defecto), red, black, ... o cualquier otro definido
% texto_subrayado: texto, perteneciente a un párrafo, que debe aparecer subrayado
% texto_al_margen: texto que aparecerá al margen en el mismo color que la línea de subrayado
\newcommand{\notamargen}[3][blue]{%
  \setulcolor{#1}%
  \ul{#2}%
  \marginalnote{%
    \textcolor{#1}{\scriptsize#3}%
  }
}
...
\begin{document}
...
...estándares ya establecidos, desarrollados para un ámbito más amplio y abierto como es el
de la \notamargen{gestión de redes de telecomunicaciones}{Por ejemplo, las arquitecturas
normalizadas SNMP y TMN}. Los diferentes capítulos que...
...
\end{document}
```

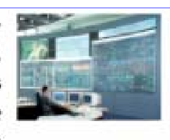
ablecidos, desarrollados para un ámbito más amplio y
de la gestión de redes de telecomunicaciones . Los difer- Por ejemplo, las arquitecturas
ie componen el documento justifican y desarrollan esta normalizadas SNMP y TMN

Icono al margen

```
...
\usepackage{calc}
...
\usepackage{graphicx}
...
\newcommand{\marginalnote}[1]{\mbox{}\marginpar{\raggedright\hspace{0pt}#1}}
```

```
% \iconomargen{alto}{fichero}
% alto: altura en número de líneas de texto a ocupar
% fichero: ruta y nombre de fichero gráfico (sólo formatos compatibles)
%          que contiene el icono a mostrar en el margen
\newcommand{\iconomargen}[2]{%
  \marginalnote{%
    \parbox[t][\baselineskip * (#1-1)][b]{\marginparwidth}{%
      \includegraphics[height=\baselineskip * (#1-1) + 1.5ex]{#2}%
    }
  }
}
...
\begin{document}
...
...\iconomargen{4}{./figs/electrica-scada}La primera parte, \nameref{sec:generalidades},
presenta los conceptos fundamentales relativos a las redes eléctricas y de ...
...
\end{document}
```

La primera parte, Generalidades, presenta los conceptos fundamentales relativos a las redes eléctricas y de telecomunicaciones. El capítulo ??, además de discutir la estructura típica de una red eléctrica, trata sobre los aspectos básicos del telecontrol clásico. El capítulo ?? se centra únicamente en la estructura de las redes de telecomunicaciones, dejando los cuestiones relativas a la gestión de las redes para la segunda parte del documento.



Creación de un índice alfabético

Hay que habilitar MakeIndex en TexnicCenter: Project -> Properties -> Uses MakeIndex (hay que cerrar y volver a abrir para que tenga efecto); con esto, pulsando "Build" se llama automáticamente a makeindex (comando que genera los índices). En el documento LaTeX hay que emplear el paquete **makeidx**, de esta manera:

```
% en el preámbulo:
\usepackage{makeidx}
\makeindex % esto habilita los comandos de makeindex
...
\begin{document}
...
\renewcommand{\indexname}{Índice alfabético} % adaptación al español
...
\index{...} % ver "The Not So Short Introduction to LaTeX"
% para más información sobre la forma de definir índices
...
\printindex % aquí se genera el índice alfabético
...
```

Escribir texto en color, subrayar en color

Se emplea el paquete **color**, del siguiente modo:

```
\usepackage{color}
...
\definecolor{rosa}{rgb}{1,0.5,0.5} % valores de las componentes roja, verde y azul (RGB)
...
\begin{document}
...
\pagecolor{black} % Color de fondo
...
\color{white} % Color del texto a partir de ese punto
...
I think I'm in \textcolor{rosa}{love} with \LaTeX\.
% "love" será escrito en rosa, el resto en el color por defecto (definido con \color)
...
\end{document}
```

Para subrayar texto de forma que la línea de subrayado esté en color, se utiliza el paquete `soul`, como sigue:

```
\usepackage{color,soul}
...
\definecolor{rosa}{rgb}{1,0.5,0.5} % valores de las componentes roja, verde y azul (RGB)
\setulcolor{rosa}
...
\begin{document}
...
El \ul{objetivo} es... % "objetivo" aparece subrayado en color rosa
...
\end{document}
```

Configuración del formato de cabecera y pie de página

Para este propósito se emplea el paquete **fancyhdr**, tal como sigue:

```
\usepackage{fancyhdr}
```

```
\pagestyle{fancy}
...
\fancyhf{} % borrar todos los ajustes

% En lo siguiente, fancyhead sirve para configurar la cabecera, fancyfoot para el pie.
% Justificación: C=centered, R=right, L=left, (nada)=LRC
% Página: O=odd, E=even, (nada)=OE
\fancyhead[RO,LE]{Cabecera1}
\fancyhead[LO,RE]{Cabecera2}
\fancyfoot[LO,CE]{Pie1}
\fancyfoot[RO,CE]{Pie2}
... (otros comandos \fancyhead o \fancyfoot)

% Modifica el ancho de las líneas de cabecera y pie
\renewcommand{\headrulewidth}{0.4pt}
\renewcommand{\footrulewidth}{0.4pt}
...
```

Si en alguna página específica queremos que se aplique un estilo concreto, usamos **\thispagestyle{*arg*}**, con *arg*={fancy | plain | empty} dependiendo de si queremos aplicar el estilo especial, el estilo por defecto o ninguno (sin cabecera ni pie), respectivamente.

Cabecera1, *Cabecera2*, *Pie1*, *Pie2*, ... pueden contener el número/nombre del capítulo/sección, etc. Para ello, hay que tener en cuenta que:

- **\leftmark** = información de nivel superior (p.e., capítulo en clase book)
- **\rightmark** = información de nivel inferior (p.e., sección en clase book)

Estos comandos se introducen en `\fancyhead` o `\fancyfoot` según se requiera, por ejemplo, en un documento clase *book*, `\fancyhead[LO,RE]{\leftmark}` indica que debe aparecer el nombre del capítulo en la parte izquierda de la cabecera si es página impar, y en la derecha si es página par. Para controlar cómo se representan los capítulos, secciones, etc., en la cabecera/pie del documento, se redefinen los comandos `\chaptermark`, `\sectionmark`, `\subsectionmark`, etc. tras la llamada a `\pagestyle{fancy}`, por ejemplo:

```
\renewcommand{\chaptermark}[1]{\markboth{\chaptername \thechapter. #1}{}}
```

donde:

- `\chaptername` = "Chapter" (por defecto) o "Capítulo" si se ha redefinido tal como se indica al principio de este documento
- `\thechapter` = número de capítulo (obsérvese que irá seguido de un punto)
- `#1` = argumento de `\chaptermark`, que es el título del capítulo

Un posible ejemplo de presentación de un capítulo en la cabecera o pie siguiendo este formato sería "Capítulo 20. Aspectos avanzados".
Es posible omitir alguno de los argumentos anteriores, encerrarlos entre comandos de formateo como `\textbf{...}` o `\MakeUpperCase{...}` (convertir a mayúsculas), etc. Aunque el ejemplo es válido para capítulos, se hace de manera análoga para secciones (`\sectionmark`, `\sectionname`, `\thesection`), subsecciones (`\subsectionmark`, `\subsectionname`, `\thesubsection`), etc.

NOTA: puede ser necesario ampliar el valor de altura de la cabecera (`\headheight`, por defecto, 12pt) o el pie (`\footskip`, por defecto 30pt). Esto nos lo indicará el propio LaTeX. Para aumentar `\headheight` a 15pt, por ejemplo, puede usarse el comando `\setlength{\headheight}{15pt}` o bien `\addtolength{\headheight}{3pt}`. También es posible aumentar o disminuir esta magnitud por un cierto factor; por ejemplo, para un incremento del 125%: `\setlength{\headheight}{1.25\headheight}`.

El número de página es **\thepage**. Puede aparecer en `\fancyhead` o `\fancyfoot`, según se quiera; por ejemplo, `\fancyfoot[C]{\thepage}` indica que el número de página aparecerá centrado en el pie de todas las páginas. El formato del número de página se establece con **\pagenumbering{*arg*}**, donde *arg*: {**arabic** = números árabes | **roman** = números romanos en minúscula | **Roman** = números romanos en mayúsculas | **alph** = letras en minúscula | **Alph** = letras en mayúscula}.

A continuación se muestra un ejemplo completo:

```
\documentclass[...,twoside,...]{book} % Documento de clase book a dos caras
...
\usepackage{fancyhdr}
...
\pagestyle{fancy}
\fancyhf{}
\fancyhead[LO]{\leftmark} % En las páginas impares, parte izquierda del encabezado, aparecerá el nombre de capítulo
\fancyhead[RE]{\rightmark} % En las páginas pares, parte derecha del encabezado, aparecerá el nombre de sección
\fancyhead[RO,LE]{\thepage} % Números de página en las esquinas de los encabezados

\renewcommand{\chaptermark}[1]{\markboth{\textbf{\thechapter. #1}}{}} % Formato para el capítulo: N. Nombre
\renewcommand{\sectionmark}[1]{\markright{\textbf{\thesection. #1}}} % Formato para la sección: N.M. Nombre

\renewcommand{\headrulewidth}{0.6pt} % Ancho de la línea horizontal bajo el encabezado
\renewcommand{\footrulewidth}{0.6pt} % Ancho de la línea horizontal sobre el pie (que en este ejemplo está vacío)
\setlength{\headheight}{1.5\headheight} % Aumenta la altura del encabezado en una vez y media
...
\begin{document}
...
```

Encabezado por defecto:

20 CAPÍTULO 3. REDES DE TELECOMUNICACIONES

3.3. SERVICIOS OFRECIDOS POR UNA RED DE TELECOMUNICACIONES 21

Encabezado obtenido con `fancyhdr`:

20 3.2. Estructura de una red de telecomunicaciones

3. Redes de telecomunicaciones 21

Portadas con gráficos

Sustituir `\maketitle` por lo siguiente:

```
% Portada gráfica
\begin{titlepage}
  \begin{figure}
    \begin{center}
      \includegraphics[width=11cm]{portada}
    \end{center}
  \end{figure}
\end{titlepage}
```

que establece como página de portada el contenido del fichero *portada.eps/pdf/png/jpg* (dependiendo de si se usa `latex` o `pdflatex`; la extensión se busca automáticamente), con un ancho de 11cm. No hace falta incluir los comandos `\title`, `\author` ni `\date`, ya que realmente `\maketitle` nunca es invocado. La portada se puede diseñar con cualquier software gráfico, lo cual nos da libertad para proporcionarle el aspecto que queramos.

Salto de línea tras título de secciones `\paragraph`

Si nuestro documento está estructurado en muchos niveles, seguramente queramos que las secciones de nivel `\paragraph` tengan un título al mismo estilo que las secciones de más alto nivel, es decir: numeradas, que aparezcan en la tabla de contenido, etc. Para ello, se emplean los comandos `\setcounter` que aparecen en la plantilla. Sin embargo, otra característica deseable y que por defecto no se tiene es que se produzca automáticamente un salto de línea tras los títulos de las secciones `\paragraph`. Para ello hay que insertar los siguientes comandos antes de `\begin{document}`:

```
% Salto de línea tras título de secciones \paragraph
\makeatletter % necesario para que reconozca a '@' como carácter normal
\renewcommand{\paragraph}{\@startsection{paragraph}{4}{\z@}{-3.25ex \@plus
-1ex \@minus -.2ex}{1.5ex \@plus .2ex}{\normalfont\normalsize\bfseries}}
\makeatother % necesario para que restablezca '@' como carácter especial
```

Para `\subparagraph` se podría hacer algo similar.

Hiperenlaces a la web

El paquete `hyperref`, ya incluido en la plantilla anteriormente presentada, permite crear hiperenlaces en el documento resultante, que deberá ser con formato PDF. Basta simplemente con añadir lo siguiente, en cualquier punto del texto (por tanto, después de `\begin{document}`):

```
\usepackage[pdftex=true,colorlinks=true,plainpages=false]{hyperref} % Soporte hipertexto
...
\begin{document}
...
% Hiperenlace
\href{http://www.loquesea.com}{aquí}
...
```

Cambiar los márgenes

El paquete `anysize` permite modificar de forma muy sencilla los márgenes del documento, gracias al comando `\marginsize{`*margen_izq*`}{`*margen_dch*`}{`*margen_sup*`}{`*margen_inf*`}`:

```
\usepackage{anysize} % Soporte para el comando \marginsize
...
\marginsize{3cm}{2cm}{2.5cm}{2.5cm}
...
\begin{document}
...
```

Los márgenes por defecto de `anysize` (es decir, incluyendo este paquete pero sin ejecutar ningún comando `\marginsize`) son 3, 2, 2 y 2 cm, respectivamente. Nótese que los márgenes por defecto de `anysize` son distintos de los que LaTeX posee por defecto. En documentos a dos caras, las páginas pares intercambian los valores de los márgenes izquierdo y derecho para que el resultado sea consistente.

Cambiar el espacio entre líneas

El espacio entre líneas, o mejor dicho, la escala a aplicar al espacio entre líneas, se fija con el comando:

```
\baselinestretch{2.0}
```

El argumento es el factor de escala. En el ejemplo anterior, se establece un espaciado doble.

Letras capitales

Para escribir letras capitales al principio de un párrafo, ocupando varias líneas, se emplea el paquete `lettrine`, usando el comando del mismo nombre para encerrar la letra capital:

```
...
\usepackage{lettrine}
...
\begin{document}
...
\lettrine{E}l objetivo de...
...
```

Si se quiere que la letra tenga un tamaño mayor, ocupando más de dos líneas (el valor por defecto), se debe usar el parámetro lines, del siguiente modo:

```
\lettrine[lines=4]{E}l objetivo de..
```

Símbolos y caracteres raros

CTAN dispone de un documento muy útil que muestra gran cantidad de símbolos y cómo obtenerlos en LaTeX.

Incluir bibliografía en el índice (tabla de contenidos)

La solución probablemente más cómoda sea usar el paquete `tocbibind` que automáticamente hace esta tarea, simplemente incluyéndolo:

```
\usepackage{tocbibind}
```

No sólo actúa sobre el índice del documento, sino que, si usamos `pdflatex`, se actualiza también la sección "*bookmarks*" del fichero PDF resultante (cuyo contenido aparece en Adobe Acrobat Reader para facilitar la navegación por el documento).

Sin embargo, el comportamiento por defecto de este paquete pasa por incluir también referencias al índice general (tabla de contenidos), índice de tablas, índice de figuras, etc. dentro del propio índice general, lo cual puede no ser de nuestro agrado. Para evitar la inserción de estos elementos, se invoca al paquete con las opciones **`nottoc`** (no incluir el índice general), **`notlot`** (no incluir el índice de tablas), **`notlof`** (no incluir el índice de figuras), **`notindex`** (no incluir el índice alfabético), **`notbib`** (no incluir la bibliografía, aunque si estás consultando este apartado no será este el caso). Por ejemplo, para incluir todas estas secciones pero no el propio índice general:

```
\usepackage[nottoc]{tocbibind}
```

Pero, por desgracia, si el documento es de clase `book` y está subdividido en partes (`\part`), pueden aparecer problemas. Supongamos que deseamos que el documento tenga la estructura de la izquierda.

- Índice de materias
- Índice de figuras
- Índice de tablas
- Prefacio
- Parte
  - Capítulo
  - ...
- Parte
  - Capítulo
  - ...
- Parte
  - Capítulo
  - ...
- **Bibliografía**

Estructura deseada

- Índice de materias
- Índice de figuras
- Índice de tablas
- Prefacio
- Parte
  - Capítulo
  - ...
- Parte
  - Capítulo
  - ...
- Parte
  - Capítulo
  - ...
  - **Bibliografía**

Estructura obtenida

En el documento principal LaTeX tendremos lo siguiente:

```
...
\begin{document}
...

\mainmatter
...
\part{Última parte}
\chapter{...}
...
\chapter{...}
```

```
...

\backmatter
\begin{thebibliography} ... \end{thebibliography} % o bien \bibliography{referencias.bib} si usamos BiBTeX
\end{document}
```

Sin embargo, la bibliografía aparecerá como perteneciente a la última parte (estructura mostrada a la derecha). Ello se debe a que la bibliografía sigue el tratamiento de un capítulo (sin numeración), y no se ha indicado de ninguna manera el final de la última parte antes de comenzar con la bibliografía. Realmente esto no se aprecia en las páginas del índice de materias, pero si estamos trabajando con `pdflatex` se hace evidente gracias a la sección *bookmarks* de Adobe Acrobat Reader. Para solucionar el problema, haremos uso de un truco: justo antes de empezar la sección de bibliografía, se redefine el nivel de un capítulo dentro en el índice de materias con el correpondiente a una parte, así:

```
...

\backmatter
\makeatletter
  \renewcommand*{\toclevel@chapter}{-1} % hacer que chapter se ponga al nivel de part
\makeatother
\begin{thebibliography} ... \end{thebibliography} % o bien \bibliography{referencias.bib} si usamos BiBTeX
\end{document}
```

Una posible opción para ocultar el fragmento de código anterior en el cuerpo del documento podría ser redefinir el comando `\backmatter` de modo que incluya lo anterior:

```
...
\makeatletter % necesario para que reconozca a '@' como carácter normal
  \renewcommand\backmatter{ % ----- aquí empieza la definición de \backmatter según la clase book
    \if@openright
      \cleardoublepage
    \else
      \clearpage
    \fi
    \@mainmatterfalse % ----- aquí termina la definición de \backmatter según la clase book
    \renewcommand*{\toclevel@chapter}{-1} % añadimos el comando a la definición de \backmatter
  }
\makeatother
...
\begin{document}
...
\backmatter % ya no hace falta incluir explícitamente el truco
\begin{thebibliography} ... \end{thebibliography} % o bien \bibliography{referencias.bib} si usamos BiBTeX
\end{document}
```

Unidades de longitud

LaTeX reconoce las siguientes unidades de longitud:

- **in** - pulgadas (*inches*, 25.4mm)
- **mm** - milímetros
- **cm** - centímetros
- **pt** - puntos (aprox. 1/72 de pulgada)
- **em** - aproximadamente la anchura de una "M" en el tipo de letra que se esté usando
- **ex** - aproximadamente la altura de una "x" en el tipo de letra que se esté usando
- **pc** - pica (1pc = 12pt)
- **bp** - big pt (72bp = 1in)
- **dd** - didôt (1157dd=1238pt)
- **cc** - cîcero (1cc = 12dd)
- **sp** - scaled point (65536sp = 1pt)

Cambiar los títulos de partes y capítulos

El paquete `titlesec` permite, entre otras muchas cosas, modificar la forma en que se muestran los títulos de las diferentes secciones. En particular, bajo la clase `book`, probablemente queramos cambiar el aspecto de los títulos de partes (`\part`) y capítulos (`\chapter`), aunque es posible modificar cualquier otro nivel mediante `titlesec`. Dicho paquete incorpora el comando `\titleformat`, descrito a continuación. Debe tenerse en cuenta que titlesec divide el título en dos partes: la etiqueta ("Capítulo 3", "Parte IV", etc.) y el cuerpo ("Introducción", "Temas avanzados", etc.).

```
\titleformat{sección cuyo formato va a definirse: \chapter, \part}
[estilo general a usar, en nuestro caso display; consultar otras posibilidades]
{formato del título: tipo de letra, serie, etc}
{contenido de la etiqueta de sección}
{separación entre la etiqueta y el cuerpo del título; debe ser una longitud}
{código que precede al cuerpo del título; al último comando que aquí se incluya
se le pasa como parámetro el cuerpo del título}
[código que sigue al cuerpo del título]
```

Al especificar el contenido de la etiqueta o del código que precede o sigue al cuerpo del título, pueden incluirse los comandos:

- **`\titlerule[`*ancho*`]`**, que traza una línea horizontal con el ancho indicado (puede omitirse).
- **`\filleft`**, **`\filcenter`**, **`\filright`**, que fuerzan un alineamiento del texto siguiente a la derecha (rellenando la parte izquierda con espacio en blanco, de ahí el nombre), centro e izquierda respectivamente.
- **`\vspace{`*longitud*`}`** para dejar espacio en blanco verticalmente
- **`\chaptertitlename`** equivale a `\chaptername` ("Capítulo", normalmente) o `\appendixname` ("Apéndice", normalmente) según corresponda; de este modo, dentro de titleformat no hay que distinguir explícitamente ambos casos.

A modo de ejemplo:

```
...
\usepackage{titlesec}
...

\newcommand{\bigrule}{\titlerule[0.5mm]}

\titleformat{\chapter}[display] % cambiamos el formato de los capítulos
{\bfseries\Huge} % por defecto se usarán caracteres de tamaño \Huge en negrita
{% contenido de la etiqueta
 \titlerule % línea horizontal
 \filleft % texto alineado a la derecha
 \Large\chaptertitlename\ % "Capítulo" o "Apéndice" en tamaño \Large en lugar de \Huge
 \Large\thechapter} % número de capítulo en tamaño \Large
{0mm} % espacio mínimo entre etiqueta y cuerpo
{\filleft} % texto del cuerpo alineado a la derecha
[\vspace{0.5mm} \bigrule] % después del cuerpo, dejar espacio vertical y trazar línea horizontal gruesa
...
\begin{document}
...
\end{document}
```

El aspecto que se consigue con lo anterior es el siguiente:

Capítulo 1

Introducción

Para los capítulos sin numeración:

Bibliografía

Personalizar los entornos `itemize` y `enumerate`

El entorno **`itemize`** produce una lista de elementos sin numerar, estando cada uno de ellos marcado con un símbolo, que por defecto es una bola (para el primer nivel), un guión (para el segundo nivel de anidación) o una estrella (para el tercer nivel de anidación). En realidad, estos símbolos se corresponden con los valores `\labelitemi`, `\labelitemii`, `\labelitemiii`, ... para cada nivel sucesivo de iteración. Por lo tanto, para cambiar el símbolo, lo único que hay que hacer es redefinir (mediante `\renewcommand`) estos comandos, asignándoles un nuevo símbolo. Por ejemplo:

Estilo por defecto

```
\begin{itemize}
 \item Item 1
 \item Item 2
  \begin{itemize}
   \item Item 2.1
   \item Item 2.2
    \begin{itemize}
     \item Item 2.2.1
     \item Item 2.2.2
     \item Item 2.2.3
    \end{itemize}
  \end{itemize}
 \item Item 3
 \item Item 4
\end{itemize}
```

- Item 1
- Item 2
  - Item 2.1
  - Item 2.2
    - Item 2.2.1
    - Item 2.2.2
    - Item 2.2.3
- Item 3
- Item 4

Estilo personalizado

```
\renewcommand{\labelitemi}{\Laserbeam}
\renewcommand{\labelitemii}{\Radioactivity}
\renewcommand{\labelitemiii}{\Biohazard}

\begin{itemize}
 \item Item 1
 \item Item 2
  \begin{itemize}
   \item Item 2.1
   \item Item 2.2
    \begin{itemize}
     \item Item 2.2.1
     \item Item 2.2.2
     \item Item 2.2.3
    \end{itemize}
  \end{itemize}
 \item Item 3
 \item Item 4
\end{itemize}
```

- Item 1
- Item 2
  - Item 2.1
  - Item 2.2
    - Item 2.2.1
    - Item 2.2.2
    - Item 2.2.3
- Item 3
- Item 4

NOTA: muchos símbolos requieren el uso del modo matemático; en estos casos, la definición será de la forma:
`\renewcommand{\labelitemi}{$...$}`

El entorno **`enumerate`**, que produce listas numeradas, se puede personalizar de forma parecida a `itemize`. En lugar de `\labelitemi`, `\labelitemii`, ... (usaré la forma `\labelitem...i` o similar para referirme a cualquiera de ellos) aquí se dispone de los comandos `\labelenumi...i`, que especifican el formato de la etiqueta con la que LaTeX precede a cada elemento de la lista. El contador automático de cada nivel viene dado por los comandos `\theenumi...i`, que pueden ser redefinidos para que la numeración sea árabe (`\arabic{enumi...i}`), romana (`\Roman{enumi...i}` mayúsculas, `\roman{enumi...i}` minúsculas) o alfabética (`\Alph{enumi...i}` mayúsculas, `\alph{enumi...i}` minúsculas). De este modo,

```
\renewcommand{\theenumi}{\Roman{enumi}} % el contador de primer nivel usará números romanos en mayúsculas
\renewcommand{\labelenumi}{\textbf{Ejercicio} \theenumi.} % formato para primer nivel
```

Esto generará listas enumeradas con etiquetas de la forma: "**Ejercicio** I.", "**Ejercicio** II.", etc.

A continuación se muestra un ejemplo más elaborado:

Estilo por defecto

```
\begin{enumerate}
 \item Item 1
 \item Item 2
  \begin{enumerate}
   \item Item 2.1
   \item Item 2.2
    \begin{enumerate}
     \item Item 2.2.1
     \item Item 2.2.2
     \item Item 2.2.3
    \end{enumerate}
  \end{enumerate}
 \item Item 3
 \item Item 4
\end{enumerate}
```

1. Item 1
2. Item 2
   (a) Item 2.1
   (b) Item 2.2
      i. Item 2.2.1
      ii. Item 2.2.2
      iii. Item 2.2.3
3. Item 3
4. Item 4

Estilo personalizado

```
\renewcommand{\labelenumi}{%
 \textbf{\theenumi}.-
}

\renewcommand{\theenumii}{\arabic{enumii}}
\renewcommand{\labelenumii}{%
 \textbf{\theenumi}.\theenumii.-
}

\renewcommand{\theenumiii}{\arabic{enumiii}}
\renewcommand{\labelenumiii}{%
 \textbf{\theenumi}.\theenumii.\theenumiii.-
}

\begin{enumerate}
 \item Item 1
 \item Item 2
  \begin{enumerate}
   \item Item 2.1
   \item Item 2.2
    \begin{enumerate}
     \item Item 2.2.1
     \item Item 2.2.2
     \item Item 2.2.3
    \end{enumerate}
  \end{enumerate}
 \item Item 3
 \item Item 4
\end{enumerate}
```

1.- Item 1

2.- Item 2

  2.1.- Item 2.1

  2.2.- Item 2.2

    2.2.1.- Item 2.2.1
    2.2.2.- Item 2.2.2
    2.2.3.- Item 2.2.3

3.- Item 3

4.- Item 4

Listas de elementos especiales

El paquete **shortlst** permite generar listas especiales; veánse los ejemplos a continuación.

Entorno shortitemize

```
\begin{shortitemize}
 \item Estos items
 \item son muy
 \item cortos y
 \item por eso
 \item no merece
 \item la pena
 \item que
 \item ocupen
 \item líneas
 \item distintas
\end{shortitemize}
```

- Estos items
- son muy
- cortos y
- por eso
- no merece
- la pena
- que
- ocupen
- líneas
- distintas

Entorno shortenumerate

```
\begin{shortenumerate}
 \item Estos items
 \item son muy
 \item cortos y
 \item por eso
 \item no merece
 \item la pena
 \item que
 \item ocupen
 \item líneas
 \item distintas
\end{shortenumerate}
```

1. Estos items
2. son muy
3. cortos y
4. por eso
5. no merece
6. la pena
7. que
8. ocupen
9. líneas
10. distintas

Entorno runenumerate

```
A continuación haremos una demostración práctica cuyo objetivo será destacar lo siguiente:
\begin{runenumerate}
 \item presentar las características funcionales del producto,
 \item destacar sus ventajas,
 \item obviar sus inconvenientes, y por supuesto,
 \item ni mencionar siquiera todo aquello que no es capaz de hacer.
\end{runenumerate}
```

A continuación haremos una demostración práctica cuyo objetivo será destacar lo siguiente: 1. presentar las características funcionales del producto, 2. destacar sus ventajas, 3. obviar sus inconvenientes, y por supuesto, 4. ni mencionar siquiera todo aquello que no es capaz de hacer.

Tablas que ocupan más de una página

El paquete **`longtable`** permite definir, usando una versión extendida del entorno `tabular`, denominada también `longtable`, tablas que se rompen automáticamente en diferentes páginas si no caben en una sola.

El entorno longtable permite definir cuatro secciones antes de comenzar con el propio contenido de la tabla:

- Formato de la primera cabecera (firsthead): el final de la definición del formato viene marcado por el comando \endfirsthead
- Formato de las cabeceras restantes (head): el final de la definición del formato viene marcado por el comando \endhead
- Formato del último pie de tabla (lastfoot): el final de la definición del formato viene marcado por el comando \endlastfoot
- Formato de los pies de tabla restantes (foot): el final de la definición del formato viene marcado por el comando \endfoot

La definición del formato de cualquiera de estas secciones puede incluir una redefinición del título de la tabla (caption) y/o la especificación de una o más filas. A modo de ejemplo aclaratorio:

```
...
\usepackage{longtable}
...
\setlength{\arrayrulewidth}{1pt}
\setlength{\doublerulesep}{0mm}

\begin{longtable}{|p{2in}|p{1in}|c|}
\caption{Un ejemplo sencillo}\\
\hline\hline
\multicolumn{3}{|c|}{\textbf{Tabla repetitiva}}\\
\hline\hline
\multicolumn{2}{|c|}{Primera y segunda columnas} & {Tercera columna}\\
\hline
\multicolumn{2}{|c|}{p-type} & \\
\hline\hline
\endfirsthead
\caption[]{Un ejemplo sencillo (continuación)}\\
\hline\hline
\multicolumn{3}{|c|}{\textbf{Tabla repetitiva (cont.)}}\\
\hline\hline
\multicolumn{2}{|c|}{Primera y segunda columnas} & {Tercera columna}\\
\hline
\endhead
\hline
\multicolumn{3}{|c|}{Sigue $\ldots$}\\
\hline
\endfoot
\hline
\multicolumn{3}{|c|}{\textbf{Fin de la tabla repetitiva}}\\
\hline
\endlastfoot
Una línea como & ésta & 1 \\
\hline
Muchas líneas como & ésta & 2 \\
Muchas líneas como & ésta & 2 \\
...
Muchas líneas como & ésta & 2 \\
\end{longtable}
```

Para conseguir el efecto de líneas horizontales más gruesas se redefinen los valores `\arrayrulewidth` (grosor de línea de la tabla) al valor de grosor sencillo y `\doublerulesep` (separación entre dobles líneas) a 0mm, de modo que `\hline\hline` produce una única línea con un grosor doble. El resultado es el mostrado en las figuras:

Primera página **Última página**

Tabla A.1: Un ejemplo sencillo

Tabla repetitiva		
Primera y segunda columnas		Tercera columna
p-type		
Una línea como	ésta	1
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Sigue . . .		

Tabla A.1: Un ejemplo sencillo (continuación)

Tabla repetitiva (cont.)		
Primera y segunda columnas		Tercera columna
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Muchas líneas como	ésta	2
Fin de la tabla repetitiva		